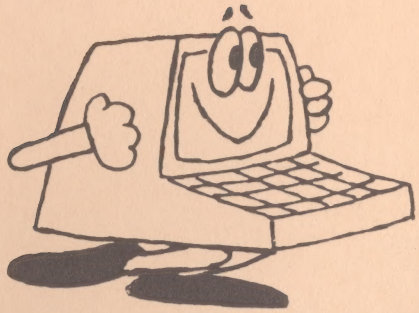

A·C·A·D·E·M·Y ON COMPUTERS

CHOOSING EDUCATIONAL SOFTWARE

There are three main categories of educational software: drill and practice, simulation, and tutorial. Drill and practice programs are those which review or drill material by rote methods. Simulation programs offer interactive learning experiences. They imitate real or imaginary situations, and allow the student to control certain variables. Tutorial programs both teach and drill. They are characterized by an ability to "branch," that is, to make decisions about the student's progress based on responses to questions. Any of these — drill and practice, simulation, or tutorial — can be presented in a game format. For this reason, educational games are sometimes considered a supra-category of the other three.

With the proliferation of educational software, it becomes increasingly important to evaluate individual programs to determine their effectiveness. Educational applications still account for only about 5% of microcomputer sales, and (perhaps understandably) commercial organizations have not yet placed a high priority on the thoughtful development of classroom-ready software. Some teachers report that as much as 80% of the software they see is inadequate in some way. Yet we know that by the end of this decade, a great deal of money will have been spent in schools on software.

Here are some questions to help you in evaluating the different types of educational software. (See also the Educational Software Evaluation Form.)

Drill and practice

- Can the student choose the number of questions he or she wishes to attempt?
- Can the student specify the level of difficulty?
- Are the instructions easy to read and understand?
- How many questions appear on the screen at one time?
- Does the program make effective use of print and graphics?
- Is the feedback the same in every instance? For example, does the program *always* say, "That's right!" in response to a correct answer?
- How does the program respond to an incorrect answer? Is the feedback encouraging or discouraging? Does the same message appear if the question is answered incorrectly a second time?
- How many times is a student allowed to attempt to answer a question? Does the program automatically give the correct answer after a specific number of tries? If so, does it also provide any explanation or instruction?
- Is the feedback for a wrong answer more interesting than the feedback for a right answer?
- Does the program test concepts and skills in a manner similar to the way in which they were presented to students?
- Can the student ask for help? Is there a "?" or "H" (help) prompt?
- Can the student get help if he or she doesn't understand the question?
- Is the student given adequate time in which to respond to the question? If time is a critical factor, is the student made aware of this at the beginning?
- Are the questions different each time the student runs the program? Do the questions always appear in the same order?
- Can the student stop at any time? Are there clear instructions on how to do this?
- Is the student given a score at the end? How much information is the student given (number of correct responses, percentage correct, time per question, overall time, etc.)?
- Can student records be run off on a printer?
- Are you (the teacher) given any additional information about the student's performance? Are you told, for example, which questions gave the student the most difficulty?
- Can the program be revised or updated easily and quickly? For example, without being a programmer, can you substitute a new list of 20 spelling words or a new set of mathematical problems?

NOTE: "Choosing Educational Software" and "The Educational Software Evaluation Form" were originally published in the *Hands-On Beginner's Manuals*, by Trudy Van Buskirk (TVOntario 1983).

A drill program should take maximum advantage of the computer's ability to individualize instruction. It should be flexible enough to suit the needs of both the most gifted student in the class and the slowest learner.

A good drill program gives the student some control: number of questions, for example, or level of difficulty. Also, the questions should be randomly generated; that is, they should appear in a different sequence every time, so that the student cannot simply memorize the answers in order.

In a good drill program, the instructions will match the reading level of the target audience and you, as the teacher, should be able to enter your own data. That is, you should be able to customize the program without having to be a programmer.

The correct answers should be reinforced more strongly and interestingly than incorrect answers. If the reverse is true, it is quite likely that students will begin to choose wrong answers just for the fun of it! Also, there should be a variety of feedback messages — enough to keep the student interested.

Students should not have to guess at the answer indefinitely; there should be some mechanism for getting help. Also, the computer should wait until the student presses the ENTER or RETURN key before processing a response — just in case the student wants to change his or her mind!

Since there won't be any worksheets to mark after the students have completed the drill, you will want to have a record of individual performances. You may also want a printout of the students' scores.

A game format may enhance a drill and practice program, but it can never compensate for poor instructional design.

Simulation

- Does the student know from the beginning exactly what is going on and what is expected of him or her?
- How many students can work on the program at one time? Are there any options that permit individuals or groups to work simultaneously?
- Can the student go back and reread the instructions without having to rerun the entire program?
- How does the student keep track of all the variables? Is this done on the computer or with pencil and paper? If the latter, is the student made aware of this at the beginning?
- Does the program make effective use of text and graphics? Is there too much or too little text? Are the graphics clear and easy to read?
- When the student chooses one option over another, does this take effect immediately? How is this shown — animated graphics, textual explanation, bar graphs, etc.?
- Are there any "fatal" paths that can end the program too quickly?
- Can the student manipulate the variables one at a time?
- How effective is the simulation at presenting the information? Could the material have been handled better in some other, non-computerized way?

Unlike drill and practice programs which are passive, simulation programs create truly interactive learning environments. Simulations are useful when real or imaginary situations cannot be experienced or recreated, except at great expense or risk. In simulation programs, there are no right and wrong answers.

A good simulation will make it clear from the start exactly what is going on and what the student is expected to do. A simulation that allows several individuals or groups to work at the same time promotes even greater interaction — this time among students. Often, students learn a great deal just by watching each other.

Students must be able to keep track of changing variables. An excellent way to use the computer is to program it to keep track of this kind of data. If, however, the student is expected to record this information using pencil and paper, he or she should be told about this right at the beginning.

Graphics should be realistic and convincing, especially if they concern scientific concepts and skills. Text should not be so lengthy that students become bored with the presentation.

Computer simulations should not be used to replace all types of hands-on activities. Rather, they should be used as a supplement. If you, as a teacher, can present the information more effectively (and less expensively) without using a computer, by all means do so.

Tutorial

- Are the instructions clear and easy to understand?
- After several unsuccessful attempts to answer a question, is the student "branched" to a new explanation?
- Does the program elaborate on certain information and concepts as it goes along?
- Is the material presented in a logical way? Does the student have some sense of where the questions are leading?
- Can the student get help? Is there a "?" or "H" (help) prompt?
- How does the program handle unusual or unexpected responses? For example, how does it handle spelling errors? Does the program automatically reject misspelled words? Or does it accept them, but present the correct spelling?

- Does the student receive a score at the end?
- Are you, the teacher, given any additional information about the student's performance?

A good tutorial program will not reject spelling errors. A student may not be a speller, but may still understand the concept being tested. So long as the program is not a spelling tutorial, students should not be penalized too harshly for incorrect spelling. Instead, the program should accept the answer *and* show the correct spelling.

In addition, you should have access to enough additional documentation to know what the teaching strategy is and how the program will lead students through the body of information. In a really good tutorial, you would probably also have input to this strategy, and be able to adapt it to suit your own needs.

Educational Software Reviews

Often teachers and administrators are placed in the position of having to order software programs without previewing them. To assist educators in their software selection process many organizations provide useful evaluations. Listed below are some of the organizations that you can write to for informative and critical reviews of educational software.

Apple Computer Clearinghouse for the Handicapped
Prentke Romich Company
R.D. 2, P.O. Box 191
Shreve, OH 44676

Apple for the Teacher
c/o Ted Perry
5848 Riddio Street
Citrus Heights, CA 95610

Boston Computer Society (BCS)
Educational Resource Exchange
Three Center Plaza
Boston, MA 02108

California Library Media Consortium
San Mateo County Office of Education
333 Main Street
Redwood, CA 94063

California Software Clearinghouse
Office of Staff Development
State Department of Education
721 Capitol Mall
Room 634
Sacramento, CA 95814

CONDUIT
P.O. Box 388
Iowa City, IA 52244

Computer Technology Task Force
Superintendent of Public Instruction
#7510 Armstrong St., W. #FG 11
Tumwater, WA 98504

Educational Insights, Inc.
150 West Carob Street
Compton, CA 90220

Educational Products Information Exchange Institute (EPIE)
P.O. Box 620
Stony Brook, NY 11790

Florida Center for Instructional Computing
College of Education
University of South Florida
Tampa, FL 33620

Helping Schools and Community Colleges to Choose Microcomputer Courseware
c/o Dr. Vicki Blum Cohen
Microcomputer Resource Center
Box 18, Teachers College
Columbia University
New York, NY 10027

Instructional Materials Division
Department of Education
State of New Mexico
Santa Fe, NM 87501-2786

JEM Research
Discovery Park
University of Victoria
P.O. Box 1700
Victoria, BC V8W 2Y2 Canada

Materials Review and Evaluation Center
North Carolina Dept. of Public Instruction
Raleigh, NC 27611

Michigan Association for Computer Users in Learning (MACUL)
33500 Van Born Road
Wayne, MI 48184

Micro Co-op
P.O. Box 432
West Chicago, IL 60185

Continued on page 6

EDUCATIONAL SOFTWARE EVALUATION FORM

(PLEASE COPY AND USE FREELY)

SOFTWARE INFORMATION (fill in or check which apply)

Program Title:

Company:

Price — cassette:
diskette:
package:

Address:

Computer Used:

Peripherals Needed or Used If Available:
- ☐ printer
- ☐ light pen
- ☐ joy stick/game paddle
- ☐ special interface
- ☐ other

Topic Area:
- ☐ math
- ☐ spelling
- ☐ reading
- ☐ language skills
- ☐ science
- ☐ social science
- ☐ computer literacy
- ☐ other

Educational Purpose:
- ☐ teaching lesson/tutorial
- ☐ drill/practice
- ☐ review
- ☐ testing
- ☐ simulation
- ☐ general skills
- ☐ record keeping
- ☐ educational game
- ☐ other

Suitable for Use:
- ☐ daily
- ☐ weekly
- ☐ continuous until mastery
- ☐ single lesson only

Usable by:
- ☐ groups
- ☐ individuals
- ☐ without teacher

Suitable Users:
- ☐ primary grade(s)
- ☐ junior grade(s)
- ☐ secondary grade(s)
- ☐ postsecondary
- ☐ special education
- ☐ teacher/other adult
- ☐ other

GENERAL RATINGS (an overall evaluation — 1 = very poor, 2 = acceptable, 3 = very good, NA = not applicable)

A. Educational Value: B. Ease of Use: C. Effective Use of Computer:

COMMENTS: _____

SPECIFIC RATINGS (1 = no, 2 = somewhat, 3 = yes, NA = not applicable)

A. EDUCATIONAL VALUE

Program Objectives:

1. Program meets an important or relevant educational need.	1	2	3	NA
2. The content is suited to computer presentation.	1	2	3	NA
3. Program is flexible enough for intended range of users.	1	2	3	NA
4. Program can be used effectively within a classroom setting.	1	2	3	NA
5. The type of program (e.g., simulation, drill and practice) is an appropriate format for the educational goal.	1	2	3	NA

COMMENTS: _____

_____

Teaching Practices:

1. Important or new material is presented in a clear and interesting way.	1	2	3	NA
2. New terms are defined or illustrated.	1	2	3	NA
3. Student actively responds, rather than just watching a display.	1	2	3	NA
4. Student corrects errors before continuing to new material.	1	2	3	NA
5. Help routines for student are available when needed.	1	2	3	NA
6. Program checks student answers to branch to new information, review old information, or adjust feedback.	1	2	3	NA

COMMENTS: _____

_____

Continued

Program Packages:

1. Package contains a coherent set of programs.	1	2	3	NA
2. Program packages follow a consistent progression of skills.	1	2	3	NA
3. Each program in a package is distinct in providing new material or stressing different skills.	1	2	3	NA
4. Instructions, response input, and feedback are consistent across programs within the package.	1	2	3	NA

COMMENTS: _____

_____

B. EASE OF USE

Instructions:

1. Teacher's documentation is clear and complete.	1	2	3	NA
2. Written instructions for student are clear and complete.	1	2	3	NA
3. Reading level, type font and display encourage reading of instructions.	1	2	3	NA
4. Instructions can be bypassed when not needed.	1	2	3	NA
5. Selection of program options is quick and simple.	1	2	3	NA
6. Other useful resource materials (e.g., demonstration kits) are available for the teacher.	1	2	3	NA

COMMENTS: _____

_____

Response Input:

1. Number and complexity of required responses is appropriate for intended user.	1	2	3	NA
2. Responses are input in a manner familiar to user (e.g., right to left for math computations).	1	2	3	NA
3. Input errors can be changed or erased easily.	1	2	3	NA
4. Inappropriate responses (e.g., letters instead of numbers) do not disrupt the program.	1	2	3	NA
5. Student is prevented from listing the program to get answers.	1	2	3	NA

COMMENTS: _____

_____

C. EFFECTIVE USE OF COMPUTER

Display:

1. Program does something that is difficult to do without the computer (or cannot be done as well).	1	2	3	NA
2. Displays are uncrowded and easily read.	1	2	3	NA
3. Graphics and animation are helpful in illustrating or accenting information, not just used as decoration.	1	2	3	NA
4. Displays have a consistent format for presenting information but are varied to add diversity to reinforcement.	1	2	3	NA
5. Sound is used to accent important information or as a reinforcement, not as a distractor.	1	2	3	NA

COMMENTS: _____

_____

Pacing:

1. User advances at own speed (e.g., uses space bar or RETURN key to go on when ready), or the program advances at an appropriate speed.	1	2	3	NA
2. Teachers can end program before its completion.	1	2	3	NA
3. Student is prevented from advancing or leaving program to escape work.	1	2	3	NA
4. Criteria for advancing to new lesson or difficulty level can be adjusted by teacher.	1	2	3	NA

COMMENTS: _____

_____

Feedback:

1. Amount and clarity of feedback is appropriate.	1	2	3	NA
2. Feedback for correct responses is reinforcing.	1	2	3	NA
3. Feedback for incorrect responses is helpful, not insulting or overly entertaining.	1	2	3	NA
4. General feedback does not encourage speed at the expense of accuracy.	1	2	3	NA
5. Useful summary feedback is provided at the end of program.	1	2	3	NA

COMMENTS: _____

Microcomputer Software and Information for Teachers (MicroSIFT)
Northwest Regional Educational Laboratory
300 SW 6th Avenue
Portland, OR 97204

Minnesota Educational Computing Consortium (MECC)
2520 Broadway Drive
St. Paul, MN 55113-5199

National Council of Teachers of Mathematics (NCTM)
1906 Association Drive
Reston, VA 20091

SOFTSWAP
c/o Ann Lathop
San Mateo County Office of Education
333 Main Street
Redwood City, CA 94063

TALMIS
115 North Oak Park Avenue
Oak Park, IL 60301

Teacher Center for Montana
215 S. 6th Street West
Missoula, MT 59801

Technical Education Research Centers, Inc. (TERC)
8 Eliot Street
Cambridge, MA 02138

THE EDUCATIONAL SOFTWARE EVALUATION FORM

*By Dr. Mary Humphrey**

In using this or any form, one should consider the software's intended user. Many programs appear to function quite well the first time a competent, cooperative adult uses them. You can learn a lot about a program by using it a second time, deliberately making mistakes, inappropriate responses, and efforts to "cheat" and get around the program's objectives. It is also a good idea to watch children (of the grade or ability level for which the program is intended) use the program with only the instructions provided or recommended by the program author. Children can crash or disrupt a program, get around its objectives, or be confused by displays in ways that may not occur to most adults. What adults find funny or cute may be insulting or embarrassingly immature for some children.

The **Educational Software Evaluation Form** is structured around the three major criteria of the Kleiman, Humphrey, and Van Buskirk article entitled "Evaluating Educational Software." There are sub-areas with sets of more specific criteria related to each of these general areas. This organization allows you to determine more easily the strengths and weaknesses of a piece of software. It is also easy to skip over sets of items that are not relevant (e.g., those items on teaching practices when evaluating programs for teacher record keeping). The form can also be used in a shortened way for quicker, less detailed evaluations.

The form has three sections, each with a separate purpose. The first section, Software Information, asks for descriptive information in several categories (e.g., what equipment is used, cost, appropriate grade levels) which can be used as headings or descriptors for entering your program evaluation into a filing or database management system. The second section, General Ratings, is an overall evaluation meant to reflect your general impression of the software as well as a summary of your ratings from the third section, Specific Ratings. The three major criteria for evaluations under General Ratings (Educational Value, Ease of Use, and Effective Use of the Computer) are broken down into sets of detailed criteria in the Specific Rating section.

The form should be used in the manner you find most helpful. You can use all or as many of the specific ratings criteria as you wish. If you do not want to fill out all of the items in a set, you can just use them as guidelines for assigning one value to the set as a whole, thus creating a "short form." It may be helpful to fill out the Specific Ratings section first, then use the averages of your ratings under each of the major criteria as the values for your General Ratings.

---

* Dr. Mary Humphrey, formerly of the Ontario Institute for Studies in Education, is an instructional designer for CDEX Corp., a California computer book and software publisher. The article she cites in this introduction to her software evaluation form is in the October 1981 issue of *Creative Computing*.